Chapitre 5

Plus court chemin dans un grahe

Algorithme de Dijkstra

Données: un graphe pondéré G=(S,A,C) et un sommet s de S.

Résultats: Les plus courts chemins de s vers tous les autres sommets.

Etape 1: Poser $B=\emptyset$, $H=\{s\}$, $\pi(s)=0$, $\pi(x)=\infty\ \forall x\in S-\{s\}$

α le dernier sommet introduit dans H.

Etape 2: Examiner tous les arcs e dont l'extrémité initial est égale à α et l'extrémité terminale y n'appartient pas à H.

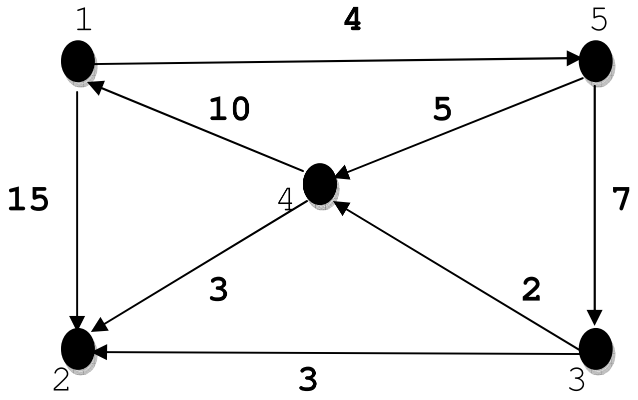
Si $\pi(\alpha)+C(e)<\pi(y)$ on pose $\pi(y)=\pi(\alpha)+C(e)$

Etape 3: Choisir un sommet $z\notin H$ tel que

$\pi(z)=$ Min $\{\pi(y), z\notin H\}$ et faire $H:=H\cup\{z\}$, $B:=B\cup\{(x,z)\}$, $x\in B$

Si H=S terminer. B définit l'arborescence des plus cours chemins issus de s. Sinon faire $\alpha=z$ et revenir à l'étape 2.

Exemple:



$H=\{1\}$, $B=\emptyset$, $\alpha = 1$.

Sommet x	1	2	3	4	5
$\pi(x)$	0	∞	∞	∞	∞

On examine les deux arcs (1,2) et (1,5):

$\pi(1)+C(1,2)=15<\pi(2)= \infty \Rightarrow \pi(2)=15$

$\pi(1)+C(1,5)=4<\pi(5)= \infty \Rightarrow \pi(5)=4$

Sommet x	1	2	3	4	5
$\pi(x)$	0	15	∞	∞	4

$\pi(5)=\min\{\pi(2),\pi(3),\pi(4),\pi(5)\}$

$\alpha=5$, $H=\{1,5\}$, $B=\{(1,5)\}$.

On examine les deux arcs (5,3) et (5,4):

$\pi(5)+C(5,3)=11<\pi(3)= \infty \Rightarrow \pi(3)=11$

$\pi(5)+C(5,4)=9<\pi(4)= \infty \Rightarrow \pi(4)=9$

Sommet x	1	2	3	4	5
$\pi(x)$	0	15	11	9	4

$\pi(4)=\min\{\pi(2),\pi(3),\pi(4)\}$

$\alpha=4$, $H=\{1,5,4\}$, $B=\{(1,5),(5,4)\}$.

On examine l'arc (4,2):

$\pi(4)+C(4,2)=12<\pi(2)= 15 \Rightarrow \pi(2)=12$

Sommet x	1	2	3	4	5
$\pi(x)$	0	12	11	9	4

$\pi(3)=\min\{\pi(2),\pi(3)\}$

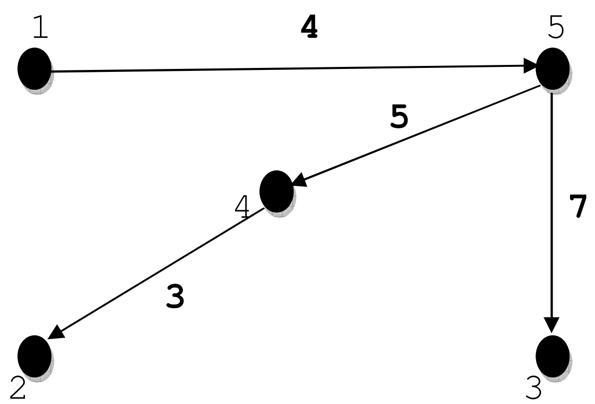
$\alpha=3$, $H=\{1,5,4,3\}$, $B=\{(1,5),(5,4),(5,3)\}$.

On examine l'arc (3,2):

$\pi(3)+C(3,2)=14>\pi(2)= 12$

$\alpha=2$, $H=\{1,5,4,3,2\}$, $B=\{(1,5),(5,4),(5,3),(4,2)\}$.

L'arborescence des plus courts chemin issus de 1 est



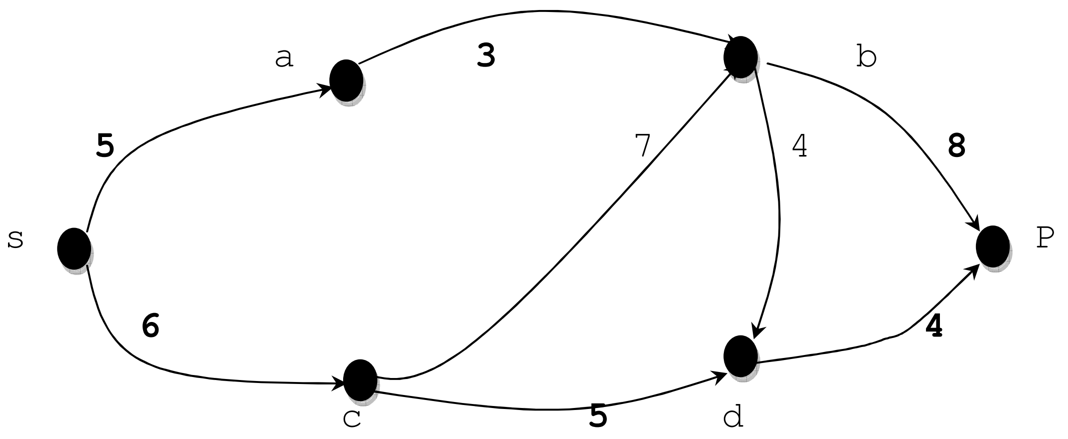
Chapitre 6

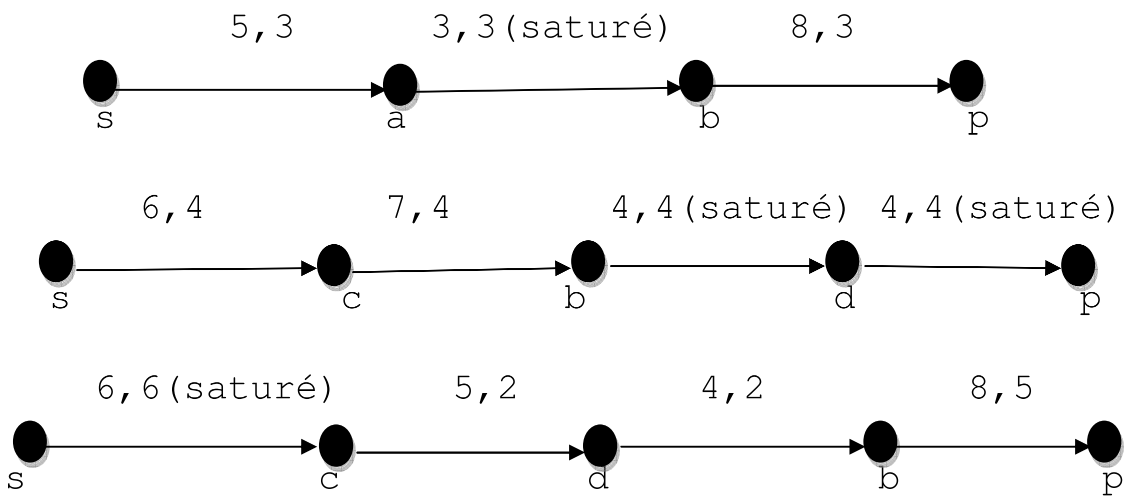
Problème du flot maximum

Le problème de flot maximum consiste à trouver la quantité maximum de flot à acheminer de la source s vers le puits p, en tenant compte des capacités de transport et de la quantité disponible en s.

Algorithme de Ford et Fulkerson

Augmenter à chaque étape la capacité d'un chemin de s vers p jusqu'à saturation





Le flot maximum = 5+4=9

Chapitre 7

Ordonnancement d'un projet

La méthode PERT

Exemple:

Un projet est composé de 6 tâches dont les durées de réalisation et les contraintes d'antériorité (de précédence) sont données dans le tableau suivant:

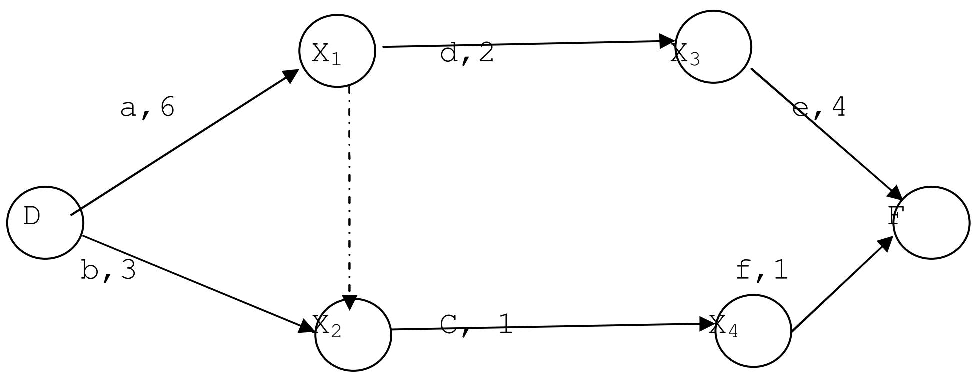
Tâche	Tâches précédentes	Durée de la tâche
a	--	6
b	--	3
c	b , a	1
d	a	2
e	d	4
f	c	1

Déterminer les dates de démarrage de chaque tâche qui minimisent le temps total de réalisation du projet.

1. **Réseau PERT associé**

   PERT (Program Evaluation and Review Technique)

   Soit R=(S,A,C) un graphe pondéré, Un arc correspond à une tâche, la valeur d'un arc représente la durée d'une tâche et un sommet est un événement signifiant le début ou la fin d'une ou plusieurs tâches.



2. Les dates:

2.1 Date au plut tôt d'une événement x:

$t_x=\max\ \{t_y+C(y,x) \mid y \in \boldsymbol{\Gamma_G^-(y)}\}$

événement	D	X_1	X_2	X_3	X_4	F
t	0	6	6	8	7	12

2.2. Date de début au plus tôt d'une tâche a: $T_a=t_x$

Tâche	a	B	C	d	e	f
T	0	0	6	6	8	7

2.3. Date au plut tard d'une événement x:

$t_F{}^*=t_F$ où F est l'événement finale

$\boldsymbol{t_x}^*=\{t_y{}^*-C(x,y) \mid y \in \boldsymbol{\Gamma_G^+(y)}\}$

événement	F	X_4	X_3	X_2	X_1	D
t*	12	11	8	10	6	0

2.4. Date de début au plut tard d'une tâche a:

$T_a{}^*=t_y{}^*-C(x,y)$

Tâche	f	E	D	c	b	a
T*	11	8	6	10	7	0

3. Tâche critique

Une tâche i est dite critique si $T_i{}^*-T_i=0$.

Tâche	**a**	B	C	**d**	**e**	f
T*-T	**0**	7	4	**0**	**0**	4

4. Ordonnancement du projet: